Drew Fielding
SEIS 765-01
Professor Howard
Due: 03/19/2025

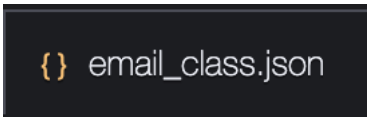
Assignment #2

1. Clone the ml_ops_sentiment_lab repo:

Cloned using git clone and the url link.

2. Remove the hard coded classes and change them to ones that are loaded from a file

Removed python list of classes and created a json file called “email_class.json”.



3. Allow users to add new classes via API. This should update the class file.

Defined a new function called load_classes()

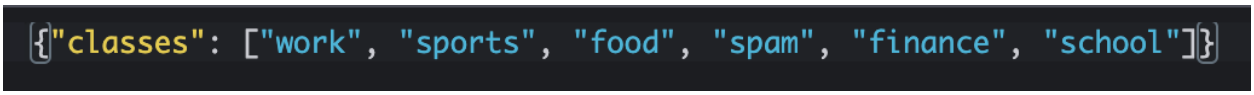
```
def load_classes():
    with open('email_class.json', 'r') as file:
        data = json.load(file)
    return data['classes']
```

Created an endpoint called add_class to accept POST requests. It will return an error message if class already exists.

```python
@app.route('/api/v1/add_class', methods=['POST'])
def add_class():
    # Get the class name from the request
    new_class = request.json.get('class_name')

    # Validate that a class name is provided
    if not new_class:
        return jsonify({'error': 'No class name provided'}), 400

    # Load current classes
    classes = load_classes()

    # Check if the class already exists
    if new_class in classes:
        return jsonify({'error': f'Class "{new_class}" already exists'}), 400

        # Add the new class and update the classes file
    classes.append(new_class)
    update_classes(classes)

    return jsonify({'message': f'Class "{new_class}" added successfully', 'classes': classes}), 200
```

Here is an example of it working and appending two new classes through curl with a post method:

```
voclabs:~/environment $ curl -X POST http://localhost:3000/api/v1/add_class -H "Content-Type: application/json" -d '{"class_name": "school"}'
{
  "classes": [
    "work",
    "sports",
    "food",
    "spam",
    "finance",
    "school"
  ],
  "message": "Class \"school\" added successfully"
}
```



I also added the capability to delete a class through Ann endpoint called delete_class to accept DELETE requests, it will provide a message of ‘not found’ if the class does not exist:

```python
@app.route('/api/v1/delete_class', methods=['DELETE'])
def delete_class():
    # Get the class name to delete from the request
    class_to_delete = request.json.get('class_name')

    if not class_to_delete:
        return jsonify({'error': 'No class name provided'}), 400

    classes = load_classes()

    # Check if the class exists
    if class_to_delete not in classes:
        return jsonify({'error': f'Class "{class_to_delete}" not found'}), 404

    # Remove the class and update the classes file
    classes.remove(class_to_delete)
    update_classes(classes)

    return jsonify({'message': f'Class "{class_to_delete}" deleted successfully', 'classes': classes}), 200
```

4. Demonstrate this working with many additional classes. Invoke classification endpoints with some different email text to show the system end 2 end.

Here are some examples of the classification working with the new classes invoked using curl:

GIT Hub Repository Link: https://github.com/drewfielding4792/SEIS765-MLOPS

```
voclabs:~/environment $ curl -X POST -H "Content-Type: application/json" -d '{"text": "Dog videos are my favorite"}' http://localhost:3000/api/v1/classify/
{
  "classifications": [
    {
      "class": "Animal",
      "similarity": 0.37120017409324646
    },
    {
      "class": "Sports",
      "similarity": 0.17769546806812286
    },
    {
      "class": "Food",
      "similarity": 0.13652747869491577
    },
    {
      "class": "Work",
      "similarity": 0.12785182893276215
    }
  ],
  "message": "Email classified"
}
voc                                                                                                                    :3000/api/v1/classify/
{
  "
    ],
    "message": "Email classified"
  }
    "similarity": 0.20181681215763092
  },
  {
    "class": "Work",
    "similarity": 0.19741488993167877
  },
  {
    "class": "Food",
    "similarity": 0.1889360398054123
  }
 ],
 "message": "Email classified"
}
```